**Documentation Technique**

|  |
| --- |
| **GSB – Appli Frais** |

# **SUIVI DU DOCUMENT**

Table des matières

1. INTRODUCTION1
   1. LANGAGE UTILISE2
2. BASE DE DONNEES4
   1. STRUCTURE DES TABLES5
      1. STRUCTURE DE LA TABLE ETAT5
      2. STRUCTURE DE LA TABLE FICHEFRAIS5
      3. STRUCTURE DE LA TABLE FRAISFORFAIT5
      4. STRUCTURE DE LA TABLE LIGNEFRAISFORFAIT5
      5. STRUCTURE DE LA TABLE LIGNEFRAISHORSFORFAIT5
      6. STRUCTURE DE LA TABLE UTILISATEUR5
      7. STRUCTURE DE LA TABLE VEHICULE5
3. MODELE LOGIQUE DE DONNEES4
4. IDE ET MODELES4
   1. MODELE DU CYCLE DE L’APPLICATION5
   2. DIAGRAMME DE CAS D’UTILISATION2
   3. MODELE VUE-CONTROLEUR2
5. MODULES5
   1. FRAIS KILOMETRIQUES5
   2. DESIGN5
   3. VALIDATION DES FICHES5
6. ANNEXE - CODAGE2
   1. CONTROLEUR → LOGIN (CONNEXION)5
   2. CONTROLEUR → TODO (LISTE)5

|  |
| --- |
| **INTRODUCTION** |

L’application va permettre d’établir une gestion plus précise et uniforme entre les entités du laboratoire. Elle devra permettre aux visiteurs d’inscrire leurs dépenses, de visualiser la prise en charge des remboursements (enregistré, validé, remboursé).

Le suivi des frais est actuellement géré de plusieurs façons selon le laboratoire d’origine des visiteurs. On souhaite uniformiser cette gestion.

L’application doit permettre d’enregistrer tout frais engagé, aussi bien pour l’activité directe (déplacement, restauration et hébergement) que pour les activités annexes évènementiel, conférences, autres), et de présenter un suivi daté des opérations menées par le services= comptable (réception des pièces, validation de la demande de remboursement, mise en paiement, remboursement effectué).

|  |
| --- |
| **LANGAGE UTILISE** |

Le langage utilisé pour la conception de l’application est le Java, HTML et CSS.

|  |
| --- |
| **BASE DE DONNEES** |

|  |
| --- |
| **STRUCTURE DES TABLES** |

### **STRUCTURE DE LA TABLE « ETAT »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **id** | char(2) | Non |  |
| libelle | varchar(30) | Oui | NULL |

### **STRUCTURE DE LA TABLE « FICHEFRAIS »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **idutilisateur** | int(4) | Non |  |
| **mois** | char(6) | Non |  |
| nbJustificatifs | int(11) | Oui | NULL |
| montantValide | decimal(10,2) | Oui | NULL |
| datemodif | date | Oui | NULL |
| idetat | char(2) | Oui | CR |

### **STRUCTURE DE LA TABLE « FRAISFORFAIT »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | Type | Null | Défaut |
| **id** | char(3) | Non |  |
| libelle | char(20) | Oui | NULL |
| montant | decimal(5,2) | Oui | NULL |

### **STRUCTURE DE LA TABLE « LIGNEFRAISFORFAIT »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **idutilisateur** | int(4) | Non |  |
| mois | char(6) | Oui | NULL |
| **idfraisforfait** | char(3) | Non |  |
| quantite | int(11) | Oui | NULL |
| idvehicule | int(3) | Non |  |

### **STRUCTURE DE LA TABLE « LIGNEFRAISHORSFORFAIT »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **id** | int(11) | Non |  |
| **idutilisateur** | int(4) | Non |  |
| mois | char(6) | Oui | NULL |
| libelle | varchar(100) | Oui | NULL |
| date | date | Non |  |
| montant | decimal(10,2) | Oui | NULL |

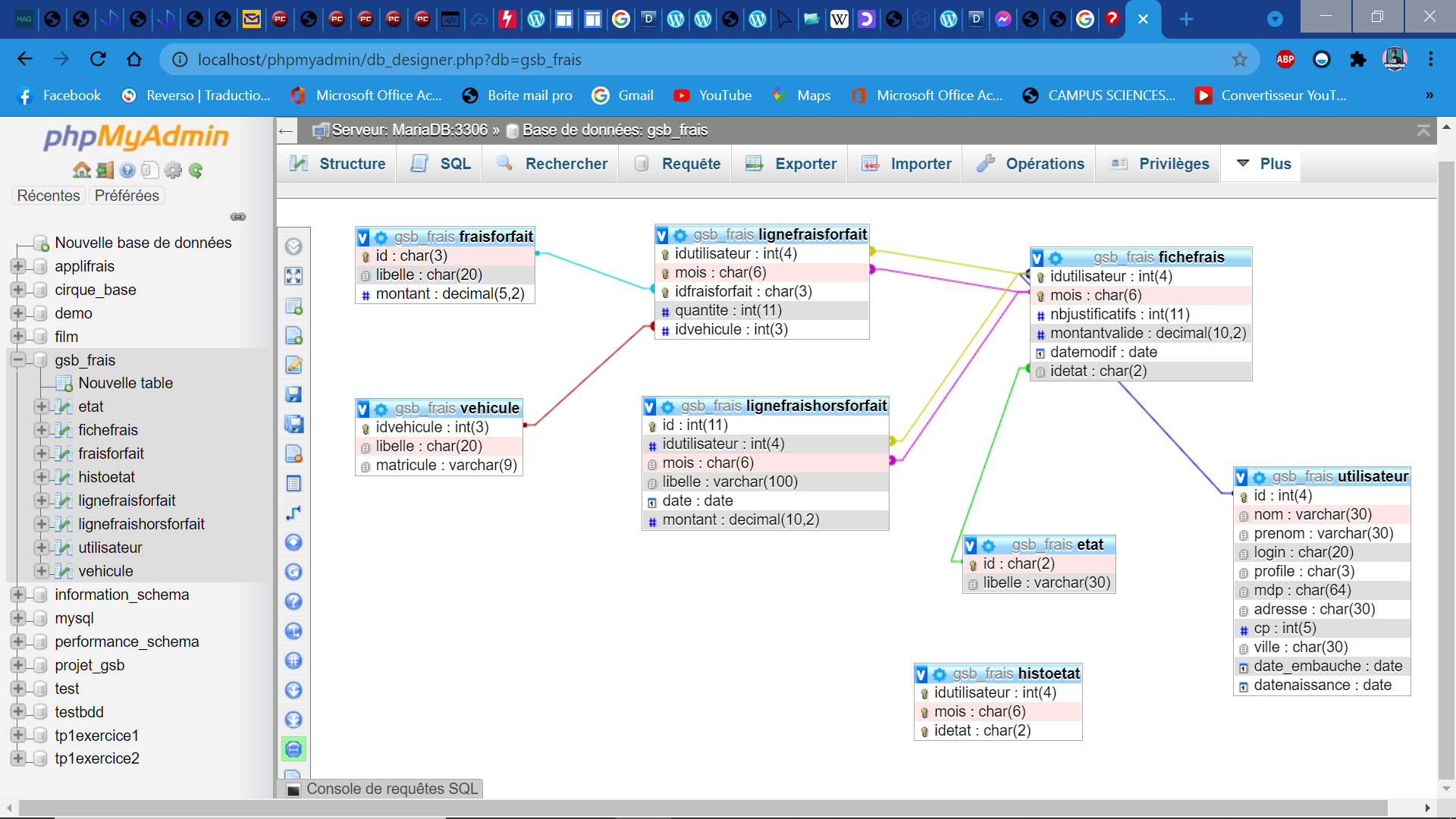
### **STRUCTURE DE LA TABLE « UTILISATEUR »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **id** | int(4) | Non |  |
| nom | varchar(30) | Oui | NULL |
| prenom | varchar(30) | Oui | NULL |
| login | char(20) | Oui | NULL |
| profile | char(3) | Oui | NULL |
| mdp | char(64) | Oui | NULL |
| adresse | char(30) | Oui | NULL |
| cp | int(5) | Oui | NULL |
| ville | char(30) | Oui | NULL |
| date\_embauche | date | Non |  |
| datenaissance | date | Non |  |

### **STRUCTURE DE LA TABLE « VEHICULE »**

|  |  |  |  |
| --- | --- | --- | --- |
| **Colonne** | **Type** | **Null** | **Défaut** |
| **idvehicule** | int(3) | Non |  |
| libelle | char(20 | Oui | NULL |
| matricule | varchar(9) | Oui | NULL |

|  |
| --- |
| **MODELE LOGIQUE DE DONNEES** |

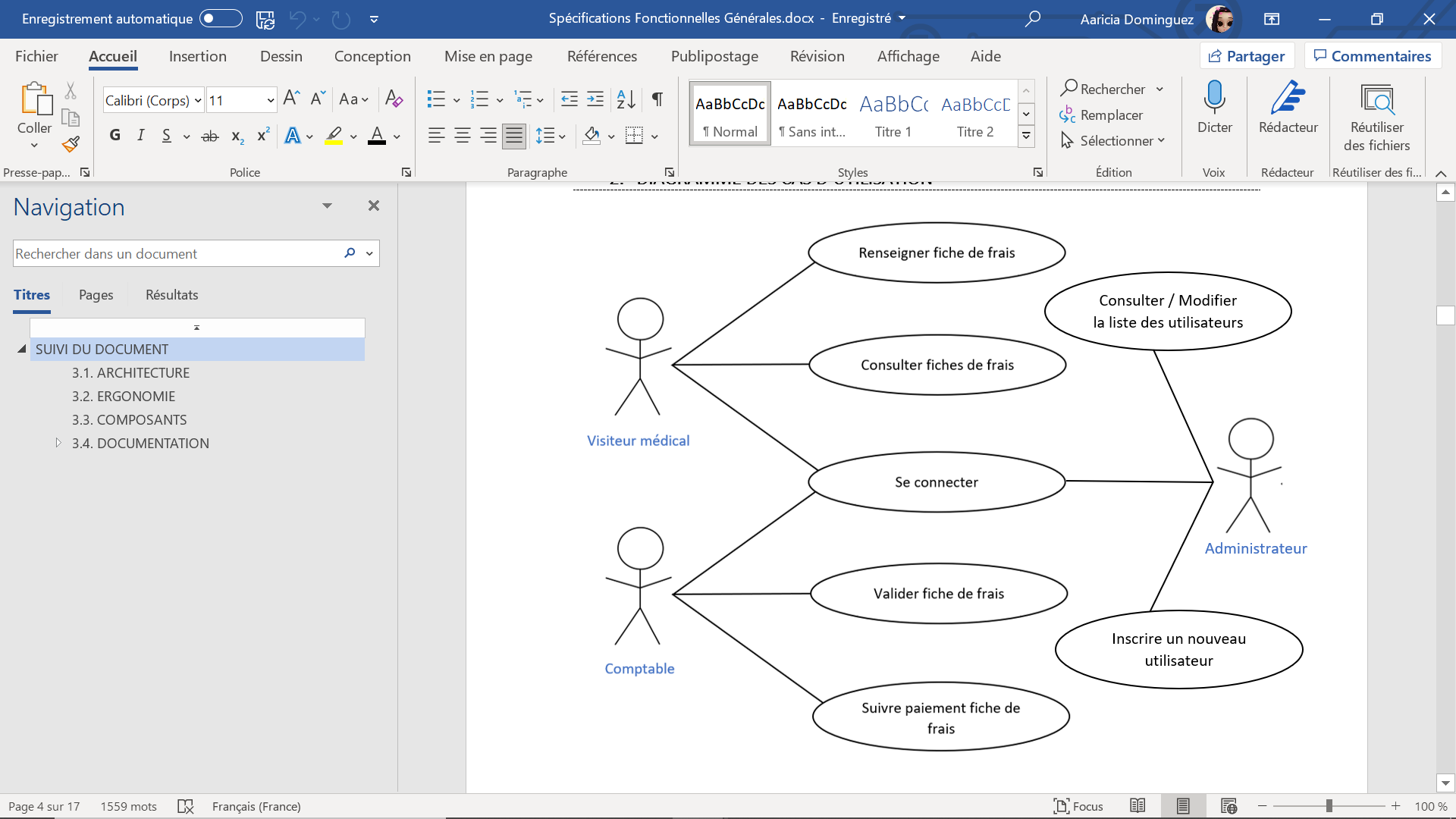


|  |
| --- |
| **IDE ET MODELES** |

|  |
| --- |
| **MODELE DU CYCLE DE L’APPLICATION** |



|  |
| --- |
| **DIAGRAMME DE CAS D’UTILISATION** |



|  |
| --- |
| **MODELE VUE-CONTROLEUR** |

Le MVC permet d’avoir une vue globale sur l’architecture de l’application.

|  |
| --- |
| **Modèle Vue-Contrôleur** |
| |  |  |  |  | | --- | --- | --- | --- | | **Client** | **Présentation** | **Contrôleur** | **Base de données** | | Visiteur    Comptable  Administrateur  Internet contour | Liste  Saisie  Login  Accueil  Connexion | Todo  Login | Engrenages contourOrdinateur contour  MySQL  Web service  JDBC | |

|  |
| --- |
| **IDE ET MODELES** |

|  |
| --- |
| **FRAIS KILOMETRIQUES** |

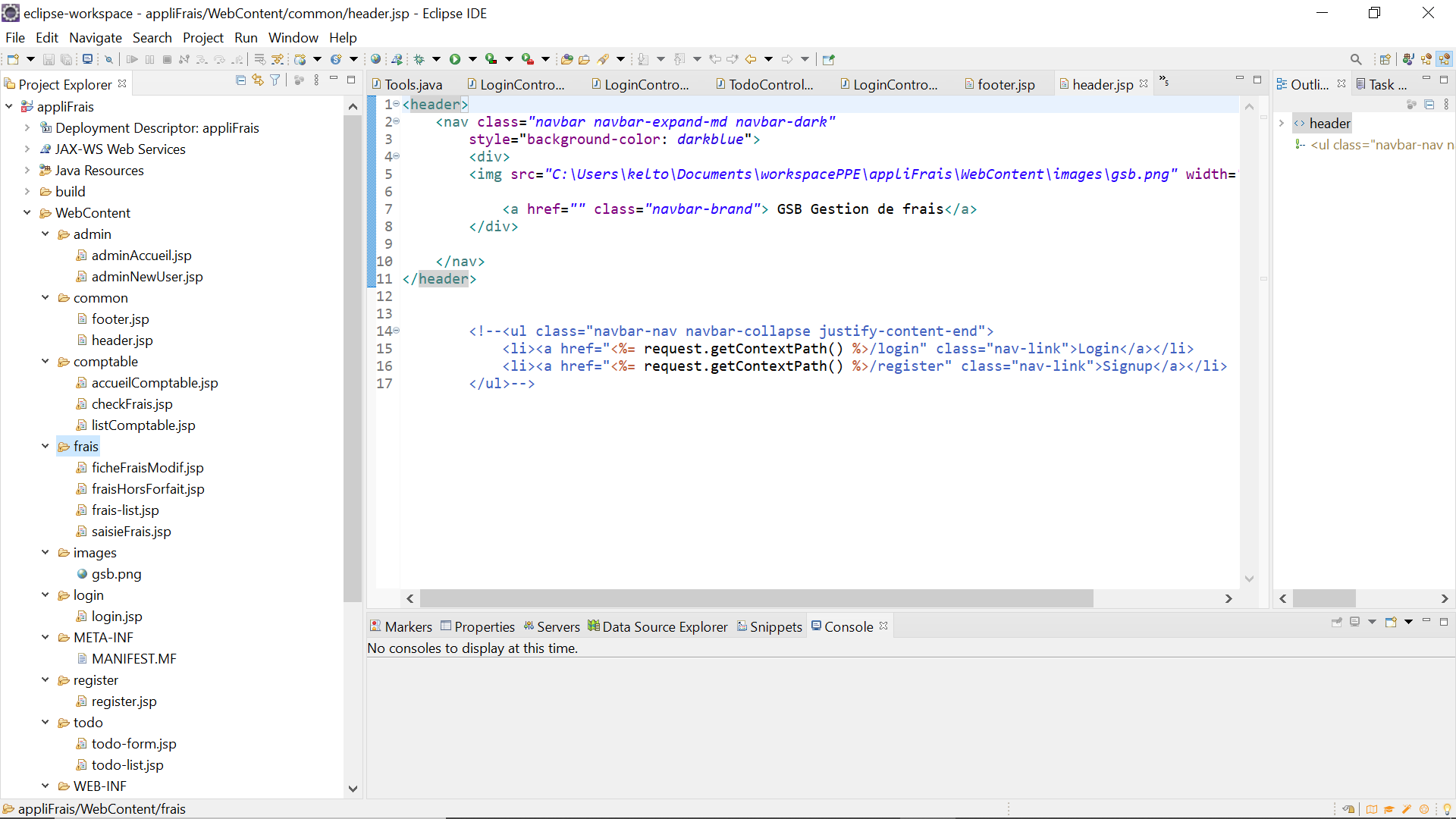
Les Frais sont enregistrés dans la table « lignefraisforfait » en fonction de leur quantité. Cependant, pour pouvoir gérer les frais kilométriques en fonction du type de véhicule utilisé, nous avons rajouté dans la saisie des frais forfaits une liste déroulante de véhicules. Ensuite, en fonction du type choisi, la quantité enregistré dans la base est directement multiplié par le ratio :

* (Véhicule 4CV Diesel) 0.52 € / Km
* (Véhicule 5/6CV Diesel) 0.58 € / Km
* (Véhicule 4CV Essence) 0.62 € / Km
* (Véhicule 5/6CV Essence) 0.67 € / Km

Lors de l’affichage des frais, on multiplie chaque quantité par le montant du frais, sauf pour les frais kilométriques, déjà transformés, qui sont multiplié par 1.

|  |
| --- |
| **DESIGN** |

L’ensemble du site est en responsive design. Nous avons utilisé Bootstrap qui intègre des fichiers CSS et Javasrcipt afin que l’application soit compatible avec un grand nombre de support.



Différentes JSP ont été créé pour simplifier et optimiser l’application. Nous avons la première JSP « Header » qui permet à l’utilisateur de s’identifier avec son identifiant et son mot de passe. Une fois connecté, les champs sont remplacés par le menu qui gère la navigation, avec l’affichage du menu « Liste de Frais » qui permet au visiteur de consulter ses fiches frais, et au comptable de les valider. Ce

Nous retrouvons la JSP « Footer » qui est le bas de la page de notre site web, avec un lien en surbrillance pour retourner à l’accueil.

Puis il y a toutes les autres JSP qui permettent la génération des autres pages web dynamiques de l’application, comme par exemple celle qui contient les éléments de saisie des fiches frais, et celle qui contient les éléments de validation des fiches frais.

Nous retrouvons le block Menu qui gère la navigation, avec l’affichage du menu « Validation » qui permet au comptable de valider les fiches des employés. Cet accès n’est pas disponible pour les autres utilisateurs.

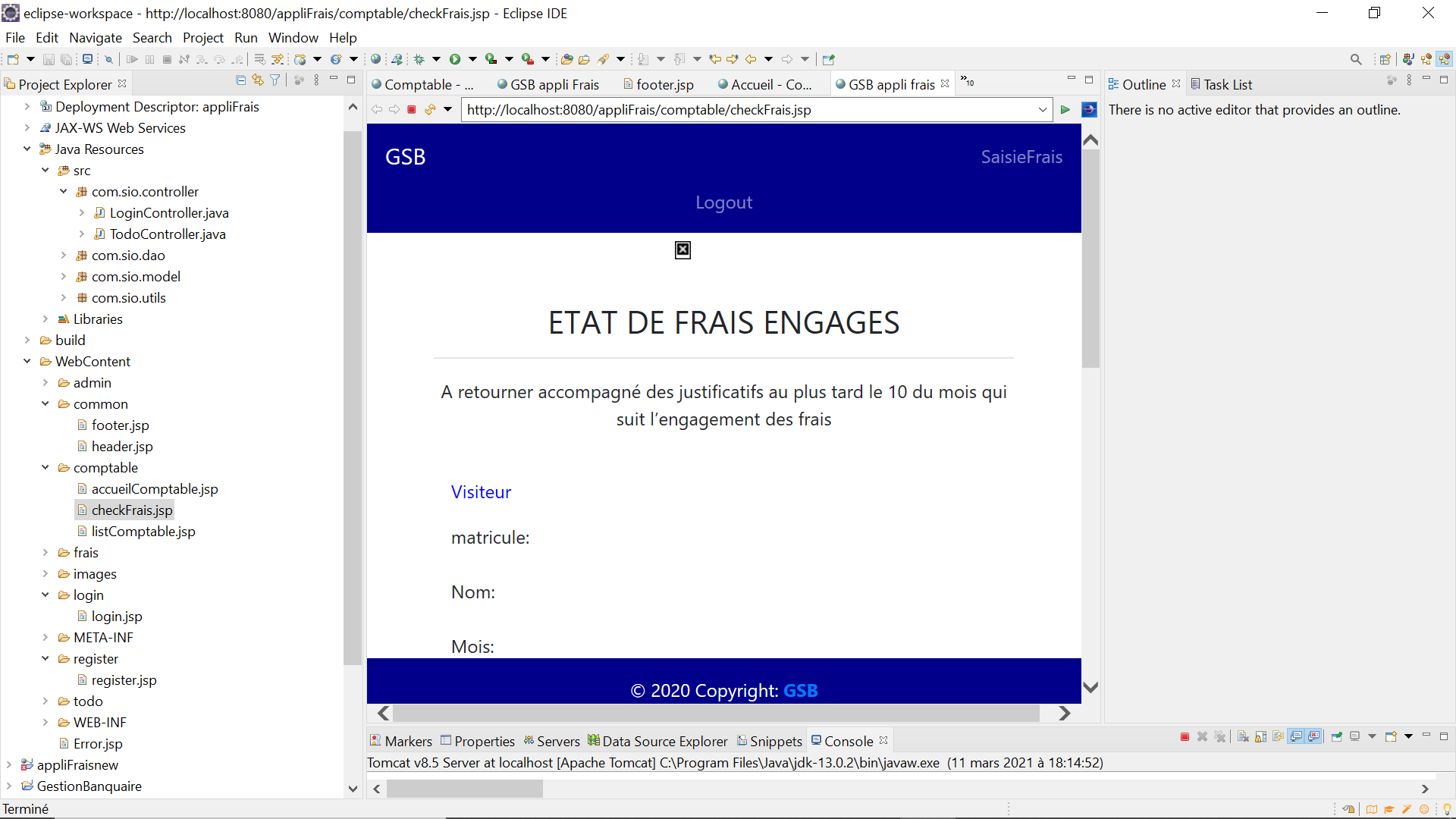
|  |
| --- |
| **VALIDATION DES FICHES** |

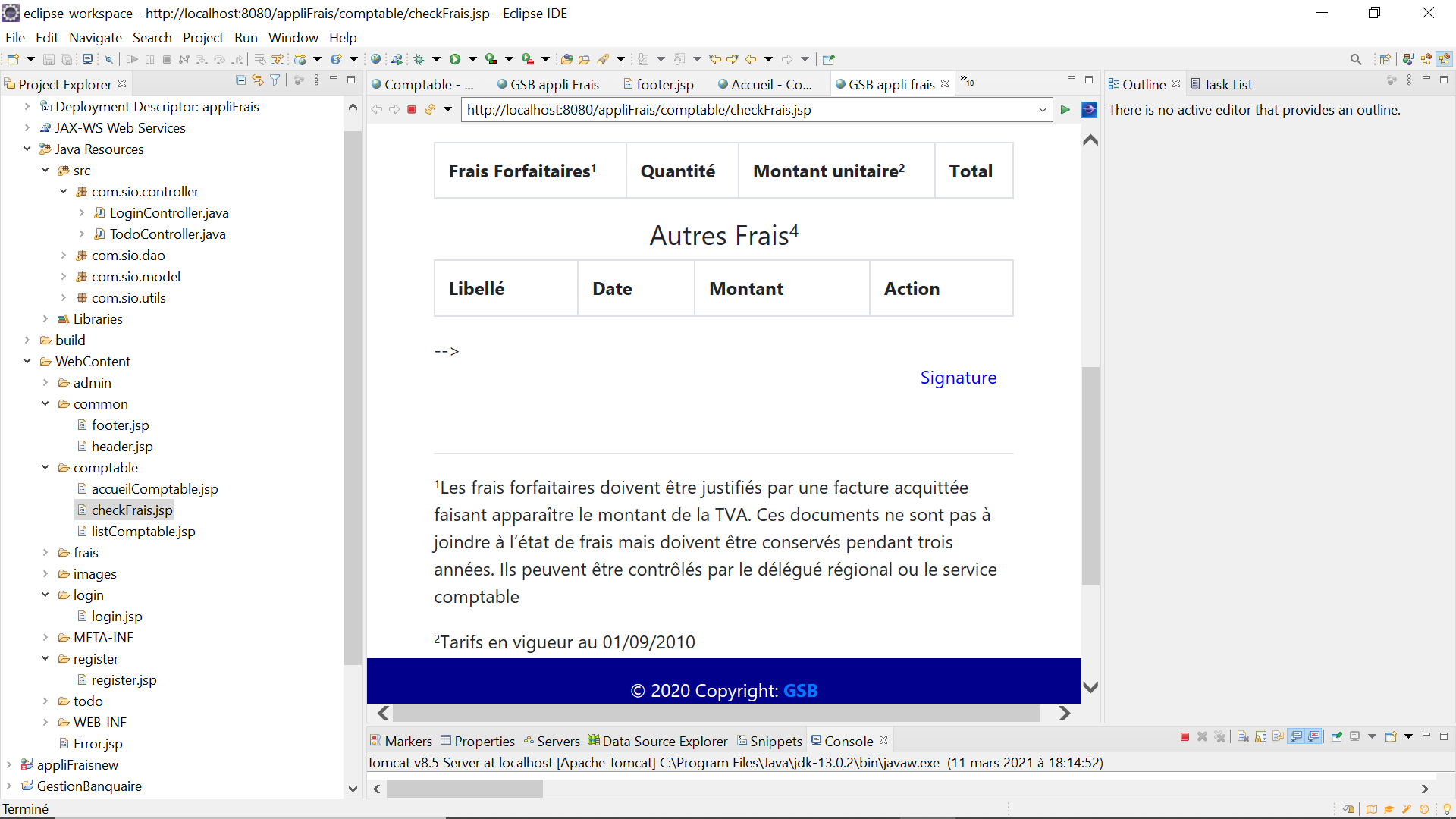
Le visiteur aura un droit d’accès à tout l’intranet. Au niveau du menu dans « Liste de frais » si le visiteur est comptable, il y aura également marqué « Liste de frais ».

Une fois que l’utilisateur aura cliqué, une page s’affiche avec une liste et une barre de recherche, avec le choix de la recherche sur le mois ou bien avec le nom d’un visiteur. Pour un visiteur lambda, une barre de recherche sur le mois sera affiché pour qu’il accède à une fiche d’un mois donné.

Si le visiteur est comptable alors deux options vont s’ajouter. Le premier est une case à cocher pour refuser ou non une ligne de frais hors forfait et la seconde est le bouton validé qui permettra de valider cette fiche.

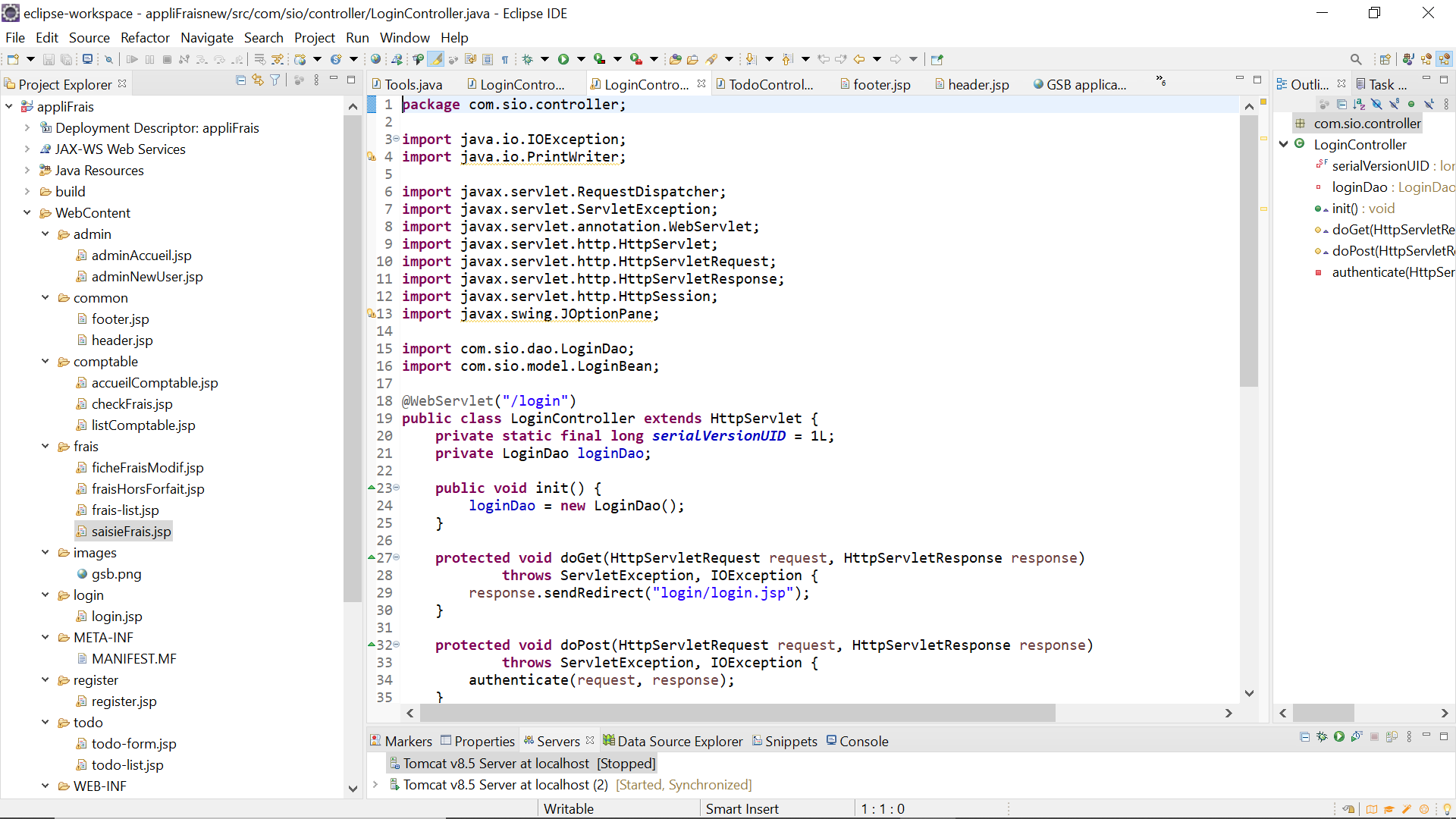
Une fois cette fiche validée, cette dernière est marquée d’un « VA » dans son suivi.

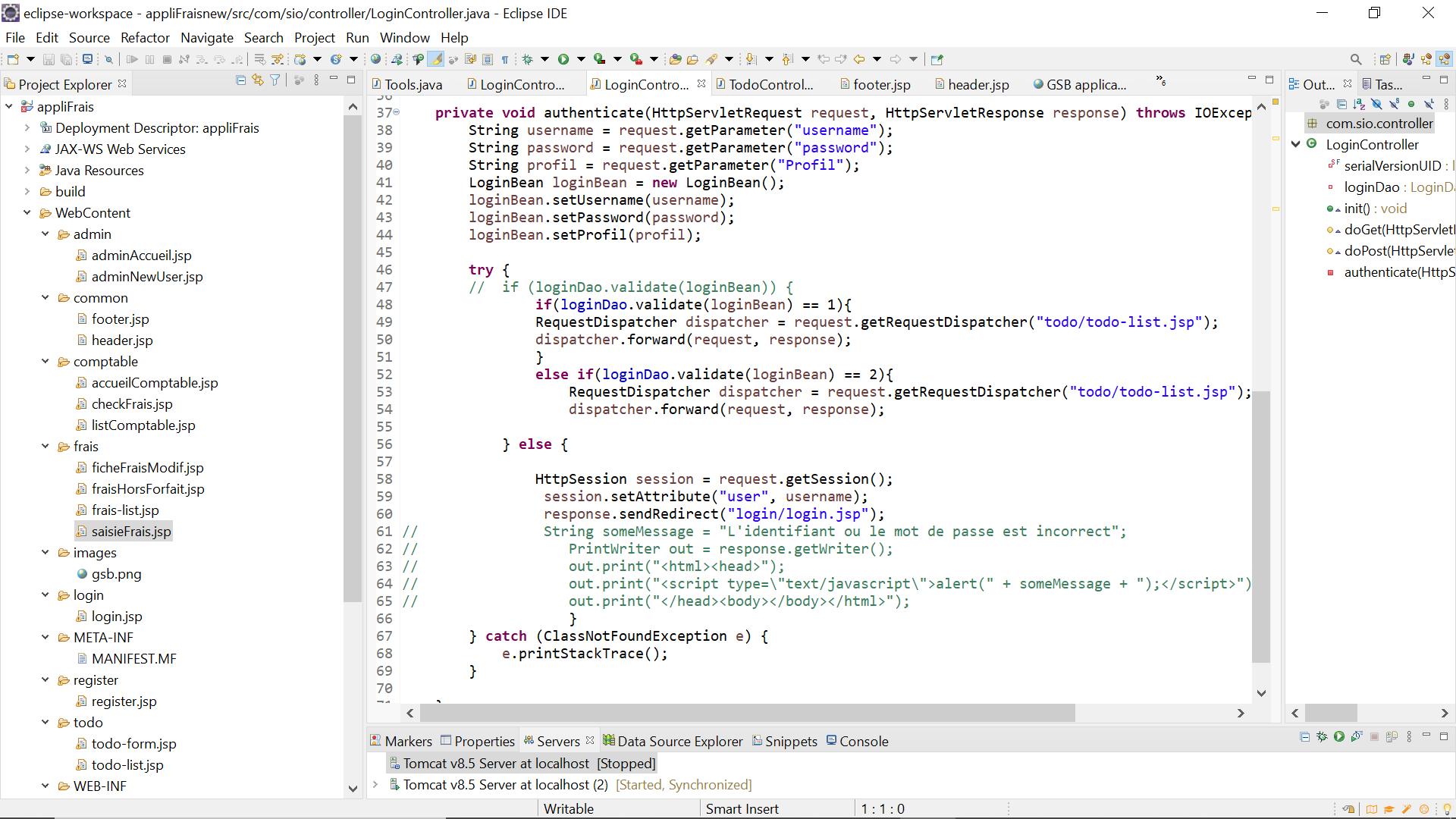




|  |
| --- |
| **ANNEXE - CODAGE** |

|  |
| --- |
| **CONTROLEUR → LOGIN (CONNEXION)** |





|  |
| --- |
| **CONTROLEUR → TODO (LISTE)** |

